WHAT IS CLAIMED IS:

1. A microcontroller comprising:

a nonvolatile memory storing instructions, each instruction having an address, the instructions constituting a first program with at least one disabled part, a second program replacing the disabled part of the first program, and an interrupt-handling routine for redirecting program execution from the disabled part of the first program to the second program;

a disabled code detector for detecting the disabled part of the first program in the nonvolatile memory and generating an interrupt signal; and

a processing unit that fetches and executes the instructions stored in the nonvolatile memory, thereby executes the first program, responds to the interrupt signal by executing the interrupt-handling routine, is thereby redirected to the second program, executes the second program, and then returns to execution of the first program.

2. The microcontroller of claim 1, wherein the disabled code detector comprises a comparator for comparing each instruction fetched by the processing unit from the nonvolatile memory with a designated instruction, the disabled part of the first program being disabled by changing at least an initial instruction in the disabled part to the designated instruction.

3. The microcontroller of claim 2, wherein the disabled code detector also comprises a counter for counting consecutive occurrences of the designated instruction, the interrupt signal being generated when the designated instruction is detected a predetermined number of times consecutively.

4. The microcontroller of claim 1, wherein the nonvolatile memory also includes a disabled address area storing an address of each disabled part in the first program.

5. The microcontroller of claim 4, wherein the disabled code detector comprises:

an address register for storing an address; and

a comparator for comparing the address stored in the address register with the address of each instruction fetched by the processing unit;

and wherein the first program includes an initialization routine causing the processing unit to load an address from said disabled address area into the address register.

6. The microcontroller of claim 1, wherein the nonvolatile memory also has an address area storing an address of said second program.

7. The microcontroller of claim 6, wherein the nonvolatile memory also stores an address decision program that refers to said address area to determine the address of said second program, the address decision program being used by the interrupt-handling routine to redirect the processing unit from the first program to the second program.

8. The microcontroller of claim 1, wherein the nonvolatile memory is an erasable programmable read-only memory.

9. A microcontroller comprising:

a nonvolatile memory storing a main program, an address decision program, an address list, and a plurality of subprograms, the subprograms including instructions for

carrying out respective tasks, the main program including instructions for controlling execution of the subprograms, the address list storing addresses of the subprograms, and the address decision program including instructions for redirecting execution from the main program to the subprograms by referring to the address list, the nonvolatile memory also having a program modification area storing at least one modified subprogram to be executed in place of one of said subprograms, the address list also storing an address of each said modified subprogram, the addresses of subprograms for which modified subprograms are provided being marked as deleted from the address list, the address decision program also including instructions for recognizing deleted addresses in the address list and reading the address of a modified subprogram from the address list in place of each deleted address; and

a processing unit that fetches and executes the instructions stored in the nonvolatile memory, thereby executes the main program and the address decision program, executes the modified subprograms stored in the program modification area, and executes the subprograms for which modified subprograms are not stored in the program modification area.

10. The microcontroller of claim 9, wherein the nonvolatile memory is an erasable programmable read-only memory.

11. A method of modifying a program stored in a nonvolatile memory in a microcontroller, the program including a main program and a plurality of subprograms executed under control of the main program, the method comprising the steps of:

(a) providing an address list in the nonvolatile memory for redirecting program execution from the main program to

the subprograms, the address list having an entry for each subprogram stored in the nonvolatile memory, said entry including an address of the subprogram and a spare address;

(b) storing a modified subprogram in an unused part of the nonvolatile memory, the modified subprogram functioning as a replacement for one subprogram among the subprograms stored in the nonvolatile memory;

(c) marking the address of said one subprogram as invalid in the entry for said one subprogram in the address list; and

(d) programming an address of the modified subprogram into the spare address in the entry for said one subprogram in the address list, whereby execution is redirected from the main program to the modified subprogram instead of being redirected from the main program to said one subprogram.

12. The method of claim 11, wherein the nonvolatile memory is an erasable programmable read-only memory.

13. The method of claim 12, wherein said step (c) comprises programming all non-programmed bits of said address.

14. The method of claim 11, wherein each said entry in the address list includes more than one spare address, whereby each subprogram can be modified more than once.